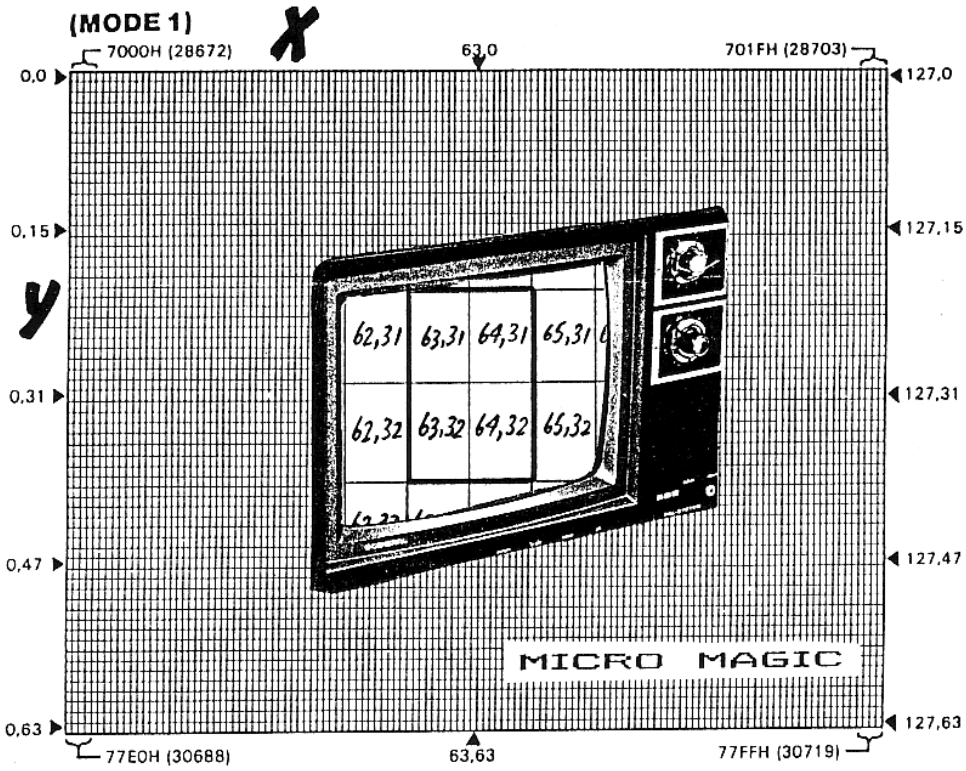
VeeZed MICRO MAGIC 5

BEGINNER'S GUIDE TO BASIC

MODE(1)



RUN and READ...

This MICRO MAGIC takes a different form.
VeeZed BASIC programmers must be familiar with the High Resolution mode, MODE(1), and their is no substitute to *"HANDS-ON"* learning.
So, instead of my writing thousands of words and confusing you, a few *example* programs should illustrate MODE(1) much better.
VeeZedders! Start your engines!

PIXELS PIXEL is an abbreviation for Picture Cell. There are 8192 of them in MODE(1). 0 to 63 (64) ACROSS and 0 to 127 (128) DOWN. See the MODE(1) Screen Grid above.

CO-ORDINATES Like a map, PIXELS are identified by co-ordinates. In the above Screen Grid you will see an X at the top (horizontal Pixels) and a Y at the side (vertical Pixels). If X=0 and Y=0 they would be the co-ordinates for the top, left-hand pixel. If X=127 and Y=63 they would be the co-ordinates for the bottom, right-hand pixel. The TV acts as a magnifying-glass for the co-ordinates of the grid in the centre of the screen.

COLOR To draw a line, a shape or a dot on the MODE(1) screen we have to change the color of the pixel(s). And this is where we go to "HANDS-ON" examples. Type in PROGRAM A.

MICRO MAGIC, 13 Brookes Street, BIGGENDEN. Qld. 4621.

A

SET(X,Y)

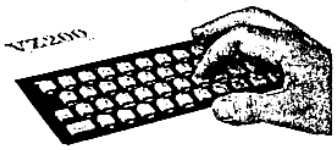
```
10 MODE(1)
20 C=1:Y=5:GOSUB100
30 C=2:Y=10:GOSUB100
40 C=3:Y=15:GOSUB100
50 C=4:Y=20:GOSUB100
60 C=5:Y=25:GOSUB100
70 C=6:Y=30:GOSUB100
80 C=7:Y=35:GOSUB100
90 C=8:Y=40:GOSUB100
92 GOTO92
100 COLORC,0:FORX=10TO50:SET(X,Y):NEXT:RETURN
```

RUN and READ...
In each of lines 20 to 90 C is the COLOR.
Y is the Vertical Co-ordinate.
In the sub-routine (line 100) the loop will draw a line across the screen. The 1st and 5th line cannot be seen because they are the same color as the backgraound.
Notice that COLORs 2, 3 and 4 are the SAME as 6, 7 and 8.
Now, change the COLOR command in line 100 to read COLORC,1.
RUN and READ...
The background has changed from GREEN to BUFF but still COLORs 2, 3 and 4 are the same as 6, 7 and 8.
Following is a Table of available COLORs:

COLOR TABLE	Pixel Color	Background Color
COLOR1,0	GREEN	GREEN
2,0	YELLOW	GREEN
3,0	BLUE	GREEN
4,0	RED	GREEN
1,1	BUFF	BUFF
2,1	LT. BLUE	BUFF
3,1	MAGENTA	BUFF
4,1	ORANGE	BUFF

Now BREAK and LIST100
Erase RETURN
Type the following line 110...

RESET(X,Y)



```
110 FORX=10TO40:RESET(X,Y):NEXT:RETURN
```

Notice that X=10TO40 whereas line 100 is X=10TO50.

RUN and READ
SET(X,Y) changes the COLOR of the Pixels.
RESET(X,Y) changes the COLOR of the Pixels back to the BACKGROUND COLOR.

POINT(X,Y)

POINT(X,Y) returns a number (1 to 4) indicating the color of a pixel.

NEW and enter PROGRAM B.

B

```
10 MODE(1)
20 COLOR7,1
30 SET(20,30)
40 A=POINT(20,30)
50 FORI=1TO1000:NEXT
60 PRINTA
70 GOTO70
```

RUN and READ
POINT(20,30) returns number 3 indicating the COLOR of the spot. Change line 20 to read COLOR3,1 and you will see why we do not use 5, 6, 7 and 8. They are the same as 1, 2, 3 and 4 anyway.

And now for the big one. Ha! Ha! There's nothing like jumping in the deep end. We'll see...

NEW
Type

```
10 MODE(1):COLOR3,0:GOSUB1000
20 COLOR2:X=0:Y=32:SET(X,Y)
30 A$=INKEY$:A$=""
40 A$=INKEY$:IFA$=""THEN40
50 IFZ$<>"B"THEN70ELSEX=X+1
60 SET(X,Y):RESET(X-1,Y)
70 A=POINT(X+1,Y)
80 IFA=3THENPRINT@103,"CRASH!"
90 GOTO40

1000 X=63:FORY=0TO63:SET(X,Y):NEXT:RETURN
```

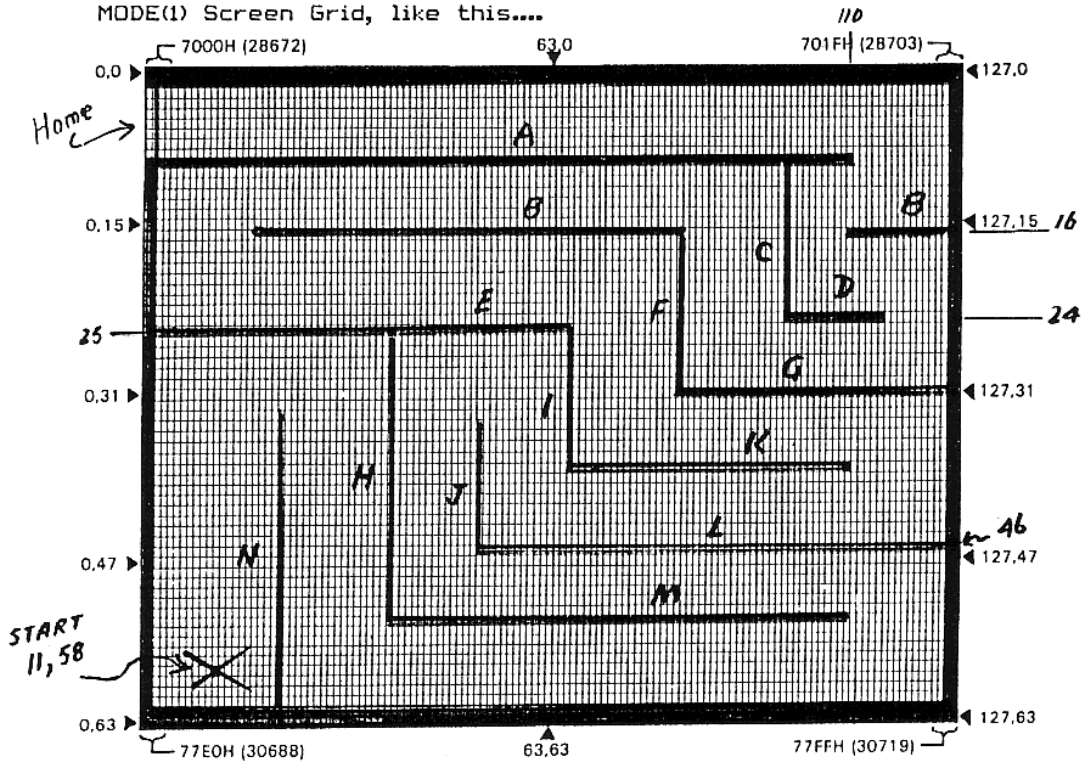
RUN and READ
After MODE(1) and COLOR3,0 sub-routine 1000 draws a line down the centre of the screen. The COLOR of the line is 3.
RETURN brings us back to line 20 where the COLOR is changed to 2 for the rest of the program. The now familiar INKEY$ causes the computer to wait until a key is pressed. Only a B is accepted. When B is pressed X is increased by 1.
Then SET(X,Y) in line 60 will put a spot next to the first one.
RESET wipes out the first one making it appear as though the spot has moved along.
A is made equal to the COLOR just ahead of the spot by POINT(X+1,Y).
Line 80 says, if the COLOR of the pixel just one place ahead of the spot is 3 then PRINT "CRASH!".
Line 90 will keep going back to line 40 until the "CRASH!".

This is a very simple example of MODE(1) MOVING GRAPHICS. Without POINT the spot would pass through the line.
Try putting SOUND31,1 in line 60. You will find out how even the shortest SOUND can slow down a program like this.

MICROMAZE

Now for a MODE(1) program. We begin with a blank screen, a blank cassette and a blank expression. Where to start? Firstly we need an idea. We have in mind a MAZE in which the player must steer a path to HOME, but if he hits the wall, or his own tail, he CRASHES. Also extra hazards will appear at random and to hit one of these is SUDDEN DEATH! Points are scored as he goes with a BONUS of 1000 if he reaches HOME.

The elements of such a program are few and do not demand a FLOWCHART, so long as we keep in mind the necessity for keeping the ACTION part of the program free from anything that would slow the action down. The INSTRUCTIONS and DRAW MAZE sub-routines will need to be AFTER the game. We plan to make the drawing of the maze begin at line 1000. The INSTRUCTIONS can begin at, say, line 1200. However, we must start by drawing the maze on a MODE(1) Screen Grid, like this....



HERE WE GO THEN...

```
100 CLS:PRINT@73,"MICROMAZE":PRINT:PRINT:Z$=INKEY$:Z$=""
110 PRINT@129,"INSTRUCTIONS? (Y OR N)"
120 Z$=INKEY$:IFZ$=""THEN120
130 IFZ$<>"N"THENGOSUB1200
140 MODE(1):COLOR,1:GOSUB1000
```

Let's get the INSTRUCTIONS out of the way

```
1199 'INSTRUCTIONS
1200 CLS:PRINT@1,"MICROMAZE.  STEER YOURSELF BY"
1202 PRINT" USING THE FOLLOWING KEYS...":PRINT
1204 PRINTTAB(12)"W = UP"
1206 PRINT:PRINT"  A = LEFT              S = RIGHT"
1208 PRINTTAB(12)Z - DOWN"
1210 PRINT:PRINT" HIT THE WALL OR DRIVE INTO"
1212 PRINT" YOUR OWN TAIL AND YOU CRASH!"
1214 PRINT" HAZARDS WILL APPEAR ON THE"
1216 PRINT" TRACK AS YOU GO. IT'S SUDDEN"
1218 PRINT" DEATH TO HIT ONE. POINTS ARE"
1220 PRINT" EARNED WITH AN EXTRA BONUS"
1222 PRINT" OF 1000 IF YOU ARRIVE HOME."
1224 PRINT"          GOOD LUCK!!!"
1230 FORT=1TO6000:NEXT:SOUND31,2:RETURN
```

In line 1206 there are 2 SPACES before A and 12 SPACES before S.

Now comes the hard part. We have to enter the sub-routine for drawing the MODE(1) maze, translating from our Screen Grid. This sub-routine will need two sub-routines of its own. One to SET the HORIZONTAL (X) maze lines, and another to SET the VERTICAL (Y) maze lines. The DRAW MAZE sub-routine is to begin at line 1000, but we'll jump ahead to lines 1100 and 1200 first to enter the drawing subs.

```
1100 FORX=ATOB:SET(X,Y):NEXT:RETURN'    HORIZONTAL LINES
1110 FORY=ATOB:SET(X,Y):NEXT:RETURN'      VERTICAL LINES
```

ENTER Temporary line: 1090 GOTO1090 to allow TEST RUNS.
Lines 1100 and 1110 will draw the maze lines depending on what values we give A, B and X for the Horizontal lines, and what values we give A, B and Y for the Vertical lines.
A and B is the length of the X lines and uses FORX=ATOB, while A and B is the length of the Y lines and uses FORY=ATOB.
OK! Let's draw the maze...

```
 999 'DRAW MAZE
1000 COLOR2:A=0:B=127:Y=0:GOSUB1100:Y=Y+1:GOSUB1100'TOP BORDER
```

RUN - Check that the TOP BORDER is correct - then LIST-1090.

```
1002 Y=62:GOSUB1100:Y=Y+1:GOSUB1100'                BOTTOM BORDER
```

TEST RUN - OK? - LIST-1090.
Continue entering and testing each line in turn.
The REM letters refer to the letters on the Screen Grid.

```
1004 A=9:B=61:X=0:GOSUB1110:X=X+1:GOSUB1110'        LEFT BORDER
1006 A=2:X=126:GOSUB1110:X=X+1:GOSUB1110'           RIGHT BORDER
1008 A=2:B=110:Y=9:GOSUB1100'                                    A
1010 A=17:B=82:Y=17:GOSUB1100:A=110:B=125:Y=16:GOSUB1100'        B
1012 A=10:B=24:X=100:GOSUB1110'                                  C
1014 A=101:B=115:Y=24:GOSUB1100'                                 D
1016 A=2:B=66:Y=25:GOSUB1100'                                    E
1018 A=18:B=30:X=82:GOSUB1100'                                   F
1020 A=82:B=125:Y=31:GOSUB1100'                                  G
1022 A=26:B=53:X=39:GOSUB1110'                                   H
1024 A=26:B=37:X=66:GOSUB1110'                                   I
1026 A=34:B=45:X=53:GOSUB1110'                                   J
1028 A=66:B=110:Y=38:GOSUB1100'                                  K
1030 A=53:B=125:Y=46:GOSUB1100'                                  L
1032 A=39:B=110:Y=54:GOSUB1100'                                  M
1034 A=33:B=61:X=21:GOSUB1110'                                   N
1036 COLOR4:FORY=2TO8:SET(0,Y):NEXT'                             HOME
1038 COLOR3:SET(11,58)
1040 RETURN
```

You will have noticed that the LEFT BORDER has A=9 leaving a gap between 2 and 8. Line 1036 fills that gap in a different COLOR. This will be used to let us know if the player reaches HOME.
Line 3038 marks the spot from where the player starts his journey.

COLOR 1 is the BACKGROUND.
COLOR 2 is the MAZE WALLS.
COLOR 3 is the MOVING SPOT.
COLOR 4 is the HOME.

ERASE Temporary line 1090.
LIST140 and ADD, after GOSUB1000 :X=11:Y=58

As mentioned earlier, there are only 4 colors. The background uses one which leaves us with only three to play with. This is limiting as we really needed another one. In the next few lines we will be responding to four events. Reaching HOME, hitting the wall, hitting one's own tail AND crashing into the random hazards. But with only three possible POINT numbers, two of the events will have to share the same response.

NOW would be a good time to CSAVE all that hard slogging, eh?

```
200 Z$=INKEY$:Z$=""
210 Z$=INKEY$:IFZ$=""THEN210
220 IFZ$="W"THENY=Y-1:GOTO300
230 IFZ$="Z"THENY=Y+1:GOTO300
240 IFZ$="A"THENX=X-1:GOTO300
250 IFZ$="S"THENX=X+1:GOTO300
260 GOTO210
```

We now have a program that RUNs but the keys for playing don't work yet....but, the next five lines will fix that...

```
300 IFPOINT(X,Y)=2THENPRINT@96,"CRASH!  YOU'RE DEAD!":GOTO500
310 IFPOINT(X,Y)=3THENPRINT@96,"YOU HIT YOUR OWN TAIL!!!":GOTO500
320 IFPOINT(X,Y)=4THENQ=1000:PRINT@96,"BONUS 1000!!!":GOTO500
330 COLOR2:A=RND(125):B=RND(61):SET(A,B):COLOR3
340 SET(X,Y):SC=SC+1:GOTO210
```

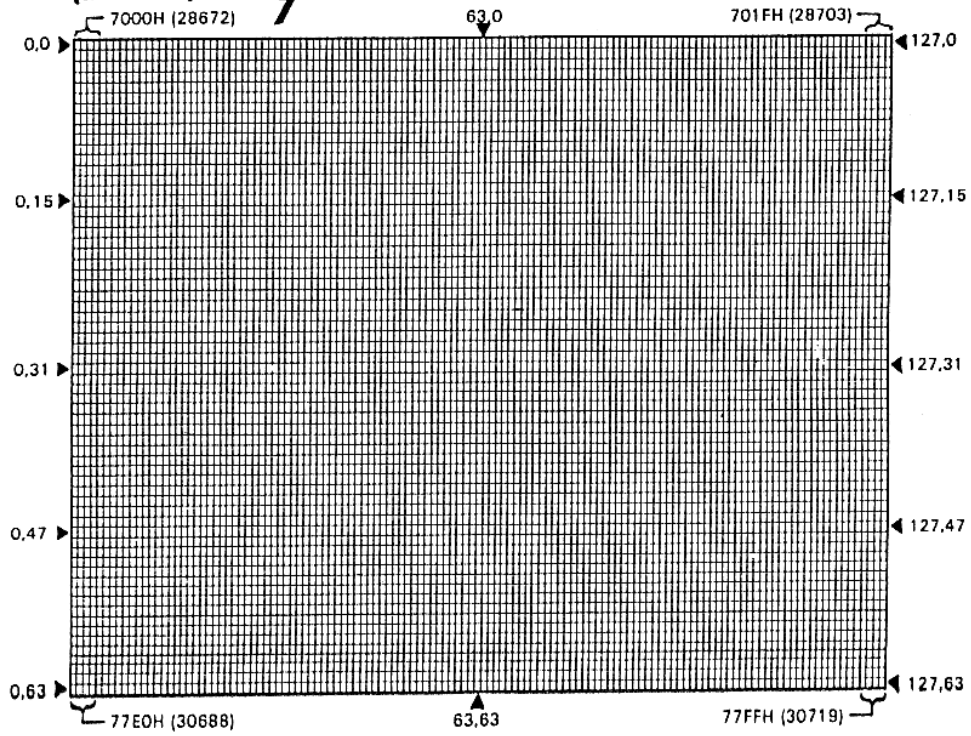
We put the checking before SET(X,Y) so that we could check the COLOR before it was changed. To check AFTER the SET would need a rather complicated system of finding out which way the player had moved. By checking the new SET before the move it MUST be correct. Agree?

Line 330 SETs RANDOM dots in COLOR2 and then puts the COLOR back to 3. We considered putting in an extra line to prevent the RANDOM dots being SET on maze lines. Something like - IFPOINT(A,B)=2THEN330 but this would have to come before the SET(A,B).
Line 340 SETs the new position and increases SC, which is the SCORE by 1 each time a move is made.

And that only leaves us from line 500 to the end

```
500 PRINT@160,"YOUR SCORE IS "SC+Q
510 Z$=INKEY$:Z$=""
520 PRINT@224,"ANOTHER GAME? (Y OR N)"
530 Z$=INKEY$:UZ$=""THEN530
540 IFZ$="N"THENCLS:END
550 IFZ$="Y"THEN140
560 GOTO530
```

(MODE 1) *Your turn now...*



More Mode (1) Jazz in Studio Ad Lib!

MICROMAZE

```
100 CLS:PRINT@73,"MICROMAZE":PRINT:PRINT:Z$=INKEY$:Z$=""
110 PRINT@129,"INSTRUCTIONS (Y OR N)"
120 Z$=INKEY$:IFZ$=""THEN120
130 IFZ$<>"N"THENGOSUB1200
140 MODE(1):COLOR,1:GOSUB1000:X=11:Y=58
200 Z$=INKEY$:Z$=""
210 Z$=INKEY$:IFZ$=""THEN210
220 IFZ$="W"THENY=Y-1:GOTO300
230 IFZ$="Z"THENY=Y+1:GOTO300
240 IFZ$="A"THENX=X-1:GOTO300
250 IFZ$="S"THENX=X+1:GOTO300
260 GOTO210
300 IFPOINT(X,Y)=2THENPRINT@96,"CRASH!  YOUR DEAD!":GOTO500
310 IFPOINT(X,Y)=3THENPRINT@96,"YOU HIT YOUR OWN TAIL":GOTO500
320 IFPOINT(X,Y)=4THENQ=1000:PRINT@96,"BONUS 1000!!!":GOTO500
330 COLOR2:A=RND(125):B=RND(61):SET(A,B):COLOR3
340 SET(X,Y):SC=SC+1:GOTO210
350 SET(X,Y)
360 GOTO210
500 PRINT@160,"YOUR SCORE IS "SC+Q
510 Z$=INKEY$:Z$=""
520 PRINT@224,"ANOTHER GAME? (Y OR N)"
530 Z$=INKEY$:IFZ$=""THEN530
540 IFZ$="N"THENCLS:END
550 IFZ$="Y"THEN140
560 GOTO530
999 'DRAW MAZE
1000 COLOR2:A=0:B=127:Y=0:GOSUB1100:Y=Y+1:GOSUB1100'TOP BORDER
1002 Y=62:GOSUB1100:Y=Y+1:GOSUB1100'              BOTTOM BORDER
1004 A=9:B=61:X=0:GOSUB1110:X=X+1:GOSUB1110'        LEFT BORDER
1006 A=2:B=61:X=126:GOSUB1110:X=X+1:GOSUB1110'     RIGHT BORDER
1008 A=2:B=110:Y=9:GOSUB1100'                                A
1010 A=17:B=82:Y=17:GOSUB1100:A=110:B=125:Y=16:GOSUB1100'B
1012 A=10:B=24:X=100:GOSUB1110'                              C
1014 A=101:B=115:Y=24:GOSUB1100'                             D
1016 A=2:B=66:Y=25:GOSUB1100'                                E
1018 A=18:B=30:X=82:GOSUB1110'                               F
1020 A=82:B=125:Y=31:GOSUB1100'                              G
1022 A=26:B=53:X=39:GOSUB1110'                               H
1024 A=26:B=37:X=66:GOSUB1110'                               I
1026 A=34:B=45:X=53:GOSUB1110'                               J
1028 A=66:B=110:Y=38:GOSUB1100'                              K
1030 A=53:B=125:Y=46:GOSUB1100'                              L
1032 A=79:B=110:Y=54:GOSUB1100'                              M
1034 A=33:B=61:X=21:GOSUB1110'                               N
1036 COLOR4:FORY=2TO8:SET(0,Y):NEXT'                         HOME
1038 COLOR3:SET(11,58)
1040 RETURN
1100 FORX=ATOB:SET(X,Y):NEXT:RETURN'        HORIZONTAL LINES
1110 FORY=ATOB:SET(X,Y):NEXT:RETURN'          VERTICAL LINES
1200 CLS:PRINT@1,"MICRO MAZE. STEER YOURSELF BY"
1202 PRINT" USING THE FOLLOWING KEYS...":PRINT
1204 PRINTTAB(12)"W = UP"
1206 PRINT"  A = LEFT            S = RIGHT"
1208 PRINTTAB(12)"Z = DOWN"
1210 PRINT:PRINT" HIT THE WALL OR DRIVE INTO"
1212 PRINT" YOUR OWN TAIL AND YOU CRASH!"
1214 PRINT" HAZARDS WILL APPEAR ON THE"
1216 PRINT" TRACK AS YOU GO. IT'S SUDDEN"
1218 PRINT" DEATH TO HIT ONE. POINTS ARE"
1220 PRINT" EARNED WITH AN EXTRA BONUS"
1222 PRINT" OF 1000 IF YOU ARRIVE HOME."
1224 PRINT"          GOOD LUCK!!!"
1230 FORT=1TO6000:NEXT:SOUND31,2:RETURN
```

MORE MODE(1) JAZZ IN STUDIO AD LIB